

We Claim:

1. A method of forestalling use of duplicate volume identifiers (volume-IDs), at least a portion of a volume being on a storage-device such that communication between an input/output (I/O) initiator and the storage-device takes place via a stack of device objects (DOs) representing the volume, the method comprising:
 - determining that a first volume-ID intended for a first volume is not available for use; and
 - restricting access to the first volume by one of
 - putting the first volume into a read-only-state in which access is restricted to read-only type access, and
 - blocking I/O to the first volume.
2. The method of claim 1, further comprising:
 - performing the step of determining before completion of a mount process for the first volume.
3. The method of claim 1, further comprising:
 - discovering that the first volume-ID is the same as a pre-existing second volume-ID in use with a second volume.
4. The method of claim 3, further comprising:
 - checking if the first volume-ID is present on a list of volume-IDs in use.
5. The method of claim 1, further comprising:
 - switching the second volume into a read-only-state in which access is restricted to read-only type access.
6. A machine-readable medium including instructions execution of which by a machine forestalls use of duplicate volume identifiers (volume-IDs), at least a portion of a volume being on a storage-device such that communication between an input/output (I/O) initiator and the storage-device takes place via a

stack of device objects (DOs) representing the volume, the machine-readable instructions comprising:

a first code segment for determining that a first volume-ID intended for a first volume is not available for use; and

a second code segment for restricting access to the first volume by one of putting the first volume into a read-only-state in which access is restricted to read-only type access, and blocking I/O to the first volume.

7. The machine-readable instructions of claim 6, wherein execution of the first code segment by the machine causes the volume-ID to be determined before completion of a mount process for the first volume.

8. The machine-readable instructions of claim 6, further comprising:

a code segment for discovering that the first volume-ID is the same as a pre-existing second volume-ID in use with a second volume.

9. The machine-readable instructions of claim 8, further comprising:

a code segment for switching the second volume into a read-only-state in which access is restricted to read-only type access.

10. The machine-readable instructions of claim 8, further comprising:

a code segment for checking if the first volume-ID is present on a list of volume-IDs in use.

11. An apparatus for forestalling use of duplicate volume identifiers (volume-IDs), at least a portion of a volume being on a storage-device such that communication between an input/output (I/O) initiator and the storage-device takes place via a stack of device objects (DOs) representing the volume, the apparatus comprising:

a memory in which is created the stack of device objects representing a storage-device, the stack including a filter device object (DO);

filter driver means for

determining that a first volume-ID intended for a first volume is not available for use, and

for restricting access to the first volume by one of

putting the first volume into a read-only-state in which access is restricted to read-only type access, and

blocking I/O to the first volume.

< Remainder of Page Intentionally Left Blank >